2018

Wi-Fi Finger-Printing Based Indoor Localization Using Nano-Scale Unmanned Aerial Vehicles

Appala Narasimha Raju Chekuri
South Dakota State University

WI-FI FINGER-PRINTING BASED INDOOR LOCALIZATION USING NANO-SCALE UNMANNED AERIAL VEHICLES

BY

APPALA NARASIMHA RAJU CHEKURI

A thesis submitted in partial fulfillment of the requirements for the

Master of Science

Major in Computer Science

South Dakota State University

2018

WI-FI FINGER-PRINTING BASED INDOOR LOCALIZATION USING NANO-SCALE UNMANNED AERIAL VEHICLES

APPALA NARASIMHA RAJU CHEKURI

This thesis is approved as a credible and independent investigation by a candidate for the Master of Science degree with major in Computer Science and is acceptable for meeting the thesis requirements for this degree. Acceptance of this thesis does not imply that the conclusions reached by the candidate are necessarily the conclusions of the major department.

Myounggyu Won, Ph.D. Date
Thesis Advisor

George Hamer, Ph.D. Date
Acting Head, EECS Department

Dean, Graduate School Date

ACKNOWLEDGMENTS

I cannot express enough thanks to my advisor Dr. Myounggyu Won of Department of Computer Science at South Dakota State University. Dr. Won always encouraged me and gave me the advice to increase the quality of my thesis. He consistently allowed my research to be my own work while continuously steering me in the right direction.

I would also like to thank the committee members for my thesis especially Dr. Sung Shin and Dr. Alireza Salehnia who gave me valuable feedback during my preliminary presentation. Their constructive comments substantially helped me in improving the quality of my thesis. I am also grateful for Dr. Suzette Burckhard for taking her time to review my thesis and provide valuable feedback during my final oral exam. Also, I would like to thank my parents. They always support me to complete learning to get Master degree.

Finally, I would like to thank the Department of Electrical Engineering and Computer Science and all the graduate faculty of the department for their continuous support and encouragement I have received during the course of my degree program. The courses I took from them helped me greatly to learn about fundamental materials that made my research activities possible.

TABLE OF CONTENTS

LIST OF FIGURES

ABBREVIATIONS

AP	Access Point
BLE	Bluetooth Low Energy
CDF	Cumulative Distribution Function
CM	Centimeter
ECN	Explicit Congestion Notification
EKF	Extended Kalman Filter
GPIO	General Purpose Input/Output
GPS	Global Positioning System
GNSS	Global Navigation Satellite System
IPS	Indoor Positioning System
MAC	Message Authentication Code
M	Meter
MAV	Micro-Scale Unmanned Aerial Vehicle
NAV	Nano-Scale Unmanned Aerial Vehicle
LED	Light Emitting Diode
LoS	Line of Sight
LQ	Link Quality
PA	Power Amplifier
PDR	Packet Delivery Rate
PID	Proportional Integral Derivative
RFID	Radio Frequency Identification

RR	Response Rate
RSS	Received Signal Strength
RSSI	Received Signal Strength Indicator
RADAR	Radio Detection and Ranging
ROS	Robotic Operating System
SSID	Service Set Identifier
TCP/IP	Transfer Control Protocol /Internet Protocol
TDMA	Time Division Multiple Access
TDoA	Time Difference of Arrival
ToA	Time of Arrival
Tx	Transmitter
UWB	Ultra-Wideband
UAV	Unmanned Aerial Vehicle
UART	Universal Asynchronous Receiver-Transmitter
URI	Uniform Resource Identifier
USB	Universal Serial Bus
VCC	Voltage at the Common Collector
WiFi	Wireless Fidelity
WLAN	Wireless Local Area Network

ABSTRACT

WI-FI FINGER-PRINTING BASED INDOOR LOCALIZATION USING NANO-SCALE UNMANNED AERIAL VEHICLES

APPALA NARASIMHA RAJU CHEKURI

2018

Explosive growth in the number of mobile devices like smartphones, tablets, and smartwatches has escalated the demand for localization-based services, spurring development of numerous indoor localization techniques. Especially, widespread deployment of wireless LANs prompted ever increasing interests in WiFi-based indoor localization mechanisms. However, a critical shortcoming of such localization schemes is the intensive time and labor requirements for collecting and building the WiFi fingerprinting database, especially when the system needs to cover a large space.

In this thesis, we propose to automate the WiFi fingerprint survey process using a group of nano-scale unmanned aerial vehicles (NAVs). The proposed system significantly reduces the efforts for collecting WiFi fingerprints. Furthermore, since these NAVs explore a 3D space, the WiFi fingerprints of a 3D space can be obtained increasing the localization accuracy. The proposed system is implemented on a commercially available miniature open-source quadcopter platform by integrating a contemporary WiFi-fingerprint-based localization system. Experimental results demonstrate that the localization error is about 2m, which exhibits only about 20cm of accuracy degradation compared with the manual WiFi fingerprint survey methods.

Chapter 1

Introduction

1.1 Motivation

Indoor localization-based services have been increasingly popular as there are more and more portable devices such as smartphones, tablets, laptops, and smartwatches consequently sparking the demand for efficient indoor localization techniques [1] [2]. Among various indoor localization mechanisms, WiFi-fingerprint-based approaches have received significant interests from industry and academia because of widespread deployment of access points (APs). This approach is non-intrusive not requiring explicit user participation (although the operator needs to put much efforts to collect WiFi fingerprints); it does not require line-of-sight communication with APs; and it has good localization accuracy in complex indoor environments [3].

WiFi-fingerprint-based indoor localization schemes have two major phases. In the training phase, WiFi fingerprints (*i.e.,* received signal strengths - RSS) of surrounding APs are collected from each reference point of a target area. Obtained RSS data are accumulated in the RSS database to train and generate a RSS model for each reference point. In the testing phase, the current location is determined by comparing measured RSS values with RSS models. Consequently, a reference point that has the closest match with measured RSS values is chosen as the current location. A major challenge is that the training phase involves a strenuous and exhaustive site survey to build the RSS database, *i.e.,* collecting a large amount of WiFi fingerprints at all reference points [1]. This time-consuming and labor-intensive process of RSS data survey has prohibited far-reaching adoption of WiFi-fingerprint-based localization systems. Furthermore, RSS signatures tend to change over time due to complex environmental factors. Thus, frequent reconstruction of the RSS database is necessary.

1.2 State of the Art

He *et al.* provide a comprehensive survey on WiFi fingerprint-based indoor localization schemes [4]. In particular, we focus on mechanisms that are developed to reduce the burden of the RSS site survey process. Crowd-sourcing has been utilized to collect WiFi RSS data [5][6][7]. However, this method is impractical in that it fails to obtain WiFi RSS data at an exact reference point because of the random mobility of participants. Thus, the quality of this type of fingerprint survey is questionable. Some approaches are based on partially labeled fingerprints. More specifically, a mapping model between estimated RSS values and geographical information of an indoor environment is developed [8][9][10]. However, these approaches may have high localization errors because they are inherently based on estimation rather than real measurements of WiFi fingerprints. To the best of our knowledge, there is few work on automating the RSS survey process to satisfy the demand for high localization accuracy.

1.3 Proposed Work

Recently, unmanned aerial vehicles (UAVs) are becoming extremely smaller and more versatile. These kinds of exceptionally small UAVs are often called the nano-scale unmanned aerial vehicles (NAVs). The small size and high maneuverability of NAVs is promising to create a variety of emerging indoor applications. In line of this new trend, in this paper, we propose an automated 3D WiFi fingerprint-based RSS data survey method utilizing NAVs. The proposed system not only reduces the time and efforts for WiFi fingerprinting but also has potential to increase the accuracy of WiFi-based indoor localization by collecting 3D fingerprint information in comparison with contemporary 2D WiFi fingerprint collection methods.

1.4 Key Contributions

The contributions of this paper are summarized as follows.

- We design, implement, and evaluate the first NAV-based indoor WiFi fingerprint survey system that significantly reduces time and efforts for collecting WiFi fingerprints.
- We perform extensive real-world experiments using NAVs to automatically collect a large amount of WiFi fingerprints.
- We present the performance analysis of the automated NAV-based WiFi fingerprinting in comparison with the manual WiFi RSS survey methods.
- We analyze the effect of 3D WiFi fingerprinting compared with traditional 2D WiFi RSS survey methods.

1.5 Thesis Organization

This thesis is organized as follows. The Chapter 1, provides the introduction with motivation, state of the art, proposed work and key contributions of our thesis. In Chapter 2, we discuss about the localization techniques in everyday human life, need for indoor localization, and several wireless indoor localization infrastructures. In Chapter 3, we review the literature of WiFi based indoor localization, UWB based positioning system, drawbacks of the existing approaches, and the issues faced by our approach. We then describe the design of the proposed system in Chapter 4 with all four components of system design. In Chapter 5, we discuss the initial approach to the work environment of RoS, Crazyflie 2.0 platform, UWB node and deck for NAV positioning, WiFi module (ESP8266). Experimental results are presented in Chapter 6. The conclusions are included in Chapter 7. The Chapter 8, provides future research direction.

Chapter 2

Related Work

In the last decade, localization techniques have grown enormously with recent advances in mobile computing technologies [11][12][13][14]. The most familiar localization application of everyday human life is navigation system using GPS localization. GPS-based localization is widely used for localization in outdoor environments. The GPS-based localization works with signals from series of GPS-satellites by using the triangulation method in order to locate the current position [15][16][17]. The GPS-based localization works more accurately when there is a clear Line of Sight (LoS) from GPS device to satellites, But in indoor environments, GPS does not perform localization as the GPS device does not have clear Line of Sight with GPS satellites and the ratio of signal to noise at GPS receiver is very close to threshold frequency limit for position detection in indoor environments [15][18][16][19]. In recent years, indoor localization techniques are innovated to achieve better accuracy for indoor environments. Most of the methods used for indoor localization are based on spatial and signal patterns, collaborative localization, motion-assisted localization and in particular spatial and signal patterns have gained most priority where this technique uses the wireless signals for localization [4]. Localization based on signal strength collection is very promising as it does not need any LoS measurements or angular measurements from AP's which leads to higher feasibility in indoor deployment [20][4][21][22].

Indoor wireless communication-based localization uses BLE, WiFi, Magnetic waves, Ultra-wideband, Zigbee, and Infrared as an infrastructure [21][20][6][23][24][7].

Rainer has surveyed thirteen different infrastructures for indoor positioning systems which include Camera-based, Infrared-based, Tactile and Combined Polar System-based, Sound-based, WLAN/WiFi-based, RFID-based, UWB-based, High Sensitive Pseudolites-based, Inertial Navigation-based, Infrastructure-based, Magnetic-based, GNSS-based and Other Radio Frequencies-Based [20]. But most of the indoor positioning techniques need LoS and can be affected by the environment which reduces the accuracy of the location.

BLE is more power efficient than WiFi and is used for short-range communications due to its limited range [25]. BLE Beacons are more flexible to deploy in the field than WiFi as they are battery powered and BLE RSS sampling has higher sampling-rate compared to WiFi RSS [26]. Bluetooth technology based indoor positioning relies on network characters like Link Quality (LQ), Received Signal Strength Indicator (RSSI) and Response Rate (RR) [25][27]. Bargh and Groote have proposed a BLE-based fingerprinting localization solution which uses the response rate (RR) of the device. This approach has approximately 97% location accuracy [27]. Chawathe has provided with a system by deploying more inexpensive Bluetooth-based beacons and used cell-based methods to determine the intersection region of visible beacon ranges [25][19]. But in both approaches, the device has to be in discoverable mode and should share the RSS signatures with other devices which have the privacy and security issue. Running a Bluetooth inquiry protocol takes around 20 seconds [19]. With short range communications and low latency of device discovery for a set of visible beacons for indoor localization has been a disadvantage for using BLE [28][19]. Li *et* *al.* have proposed a smartphone-based indoor localization with Bluetooth low energy beacons and this system uses Kalman-filter and extended Kalman-filter to reduce the noise when collecting the BLE RSS signature but has high localization accuracy error of 1.0 m to 3.1 m [26].

Zigbee was one of the low cost, low power and high accuracy approach for indoor localization [29][30]. Zigbee system was designed for short and medium range network applications which require low-power consumption but does not require large amount of data throughput [20]. The signal of Zigbee technology is open to a wide range of signal types which results in more radio communication interference and the range of Zigbee technology for an indoor system is typically around 20m to 30m [23][29]. Niu *et al.* have implemented a Zigbee assisted WiFi-based indoor localization with low power, cost-efficient and more accurate system using WiFi beacon frames and RSSI signatures to train their system [1]. Though this system was an excellent approach for indoor localization, but the user needs to quantify the WiFi RSS beacon frames to build the system which is a time taking process.

UWB-based indoor localization has higher accuracy compared to other indoor localization schemes [29][22]. UWB-based localization is used widely in short range network applications with high bandwidth communication [31][22]. Providing UWB-based indoor localization for long-range network applications suffers high probability for interference in communication between the UWB nodes, which reduces the system accuracy [31][22][20]. To provide high accuracy, the system should be designed with a lot of UWB modules in the field which significantly increases the cost of the system.

The UWB-based system needs dedicated infrastructure where WiFi-based systems utilize the building infrastructure [32][31][23]. A new approach has been proposed based on the UWB-based indoor localization for quadcopters, where a UWB-tag is attached to the quadcopter. The quadcopter uses time of flight measurements from the UWB-tag to a set of anchors in round-robin approach and based on the measurements the position of the quadcopter is estimated accurately in the field [33].

WiFi-based approaches have gained more attention in indoor localization [34][21] [35][20]. Li *et al.* have proposed an indoor localization system with several modules like step detection with a smartphone, step stride length estimation, estimation of heading direction, particle filter and personalization module for adapting step model [36]. But this system has to interact with the user for initial location through user provided input and uses the position estimation on the indoor map where the locations have to be collected manually. Liu *et al.* used the trilateration method in order to provide indoor localization based on the WiFi-based weighted screening method [21]. Trilateration uses set of three nearest WiFi access points based on the distance between the observation point and the referenced AP's. The trilateration positioning is to trace a sphere using the distance between the signal source and the observation point.

The surface of the sphere gives the equivalent signal strength of that source to an observation point anywhere on that surface. For three AP's there are three spheres which interact at a minimum of 2 points which shows a common area in Figure 2.1. With this limitation, the WiFi signal strengths may vary if there is a common area for access points which results in low accurate system [21]. The grey portion shows the common area of both AP's in Figure 2.1.

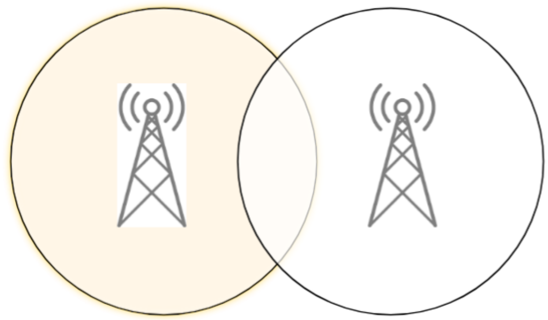

Figure 2.1: Trilateration Method: Access Point Common Area

Position uncertainty will be smaller if the common area network of both AP's is less and more position uncertainty if the common area network is higher which results in lower localization accuracy. All WiFi-based indoor localization schemes require a training data set where the data has to be collected manually [21][35][34][36].

Chapter 3

Background

3.1 Indoor Localization

Localization is a gateway to several innovative applications like navigation, people tracking, rescue operation, proximity marketing, pedestrian safety [37], traffic monitoring [38][39], and traffic management [40][41]. In the case of outdoor localization, there is a wide variety of applications like navigation which uses GPS systems in order to locate your position in the map based on longitude and latitude. But coming to indoor localization, outdoor localization techniques are not capable of providing indoor positioning services which raised a new research wing of innovative applications that help with localization in indoor environments [24]. The indoor localization or indoor positioning system (IPS) is a concept of locating objects or people using WiFi, cellular-based, infrared, Zigbee, RFID, UWB-based, Bluetooth, radio waves, and magnetic waves in an indoor environment. Due to rapid growth of mobile internet and various communication technologies, indoor localization-based services have been increasingly popular as there are more and more portable devices such as smartphones, tablets, laptops, and smart-watches [1][2]. Consequently, the demand for efficient indoor localization techniques is sparkling.

Indoor localization includes probabilistic and deterministic methods [42][30]. Probabilistic algorithms use statistical inference between the target signal measurement and the stored position. Deterministic algorithms use metrics to differentiate between the signal measurement and fingerprint data. But in indoor positioning systems, there have been numerous variables to encounter such as structural complexity of the building, the stability and coverage of the wireless signal, the accuracy of the positioning instrument, the sensitivity of the positioning instrument when measures the signal strength, and the influence of the crowd in the indoor environment [21]. Based on all these considerations, the indoor positioning systems, performance can be analyzed in terms of six different properties [43]:

- **Accuracy:** Accuracy measures the size of the gap between the actual set-point position at that considered area to the location of the same set-point estimated by the indoor positioning system which gives the error in the performance of the IPS system.

- **Complexity**: The complexity of the IPS system can be given by the amount of computing time required for the positioning algorithm. The locating rate is higher if the complexity is higher.

- **Robustness:** Good system robustness is during the positioning process, even if one of the signal sources acts abnormally or cannot be received the indoor positioning system has to still work under a certain level of accuracy.

- **Precision:** Precision is the measure between the gaps of the estimation points. The precision of this system is well explained in the chapter 6. An accurate cumulative distribution function (CDF) can be used to calculate the strengths and weaknesses, where the higher the concentration of the curves higher the precision. The stability of the signal source and the merits of the system positioning semi-autonomous flight subsystem in the section 4.3 are the main factors affecting the precision.

- **Scalability:** Scalability is the range supported by the indoor positioning system, which includes experimental setup area, signal coverage, and 2D/3D spatial positioning.

- **Cost:** System construction costs include time consumed, power consumption, and money.

Indoor localization techniques are classified into two approaches [43][44]: model-based approaches and fingerprint-based approaches.

- **Model-Based Approaches:** The model-based indoor localization techniques use geometrical models like time difference of arrival(TDoA), time of travel(ToA) for the position estimation. For this model-based approaches, we have to provide access point locations, floor plans to improve the estimation. So, establishing this model-based indoor localization is very difficult.

- **Fingerprint-Based Approaches:** Fingerprinting based localization is widely used technique, due to its performance and operating environments [45][29]. Fingerprint-based localization captures RSS signatures or fingerprints which are matched with the set-points to identify a location. The main motivation of the fingerprint-based localization is to find the set-points and use those set-points effectively to determine the location of the device, and the user who is carrying the device. Fingerprint-based localization requires a site survey to build up a fingerprint database during the offline phase. However, this fingerprinting survey process requires extensive time and efforts due to the spatial and temporal dynamics of RSSs. As such, the system maintenance cost becomes prohibitively high. This is the key motivation of this research for developing a solution to mitigate the overhead for fingerprint site survey.

3.2 Fingerprint-Based Localization

Vo and De have provided various types of fingerprint-based localization methods which include Visual Fingerprints, Motion Fingerprints, and Signal Fingerprints [46].

- **Visual Fingerprints:** In visual fingerprint based indoor localization techniques, many content-based image retrieval techniques have been proposed to search a query based image from a large image database using visual features appearing in the images like color, texture, and shape [47][46][48][29]. Mobile devices are equipped with the powerful image and video processing techniques and images taken by a mobile device can be used to point the location of the device. When the user clicks an image with her smartphone camera, visual fingerprint system uses the image to find similar images from the database based on the best-matched image with the geological location information [20][46][26].

- **Motion Fingerprint:** Today's smartphones can perform sensing and user motion recognition in the real-time with the support of motion sensors such as accelerometers and electronic compasses or gyroscope [48][49][46]. The basic idea of motion fingerprint is to combine the accelerometer and compass readings from the compass and match with the area of interest to estimate the location. If the user travels for a distance, the traveled distance can be measured periodically and used as fingerprints for localization and tracking assistance [50][46][20].

- **Signal Fingerprint:** Both mobile devices and wireless communication networks have together encouraged several types of techniques that detect wireless signals for localization [26][29][51][46]. This localization has been proposed based on the signal fingerprinting, time of arrival, the angle of arrival, and time difference of arrival [31][32][52][47]. Among them, signal fingerprinting based techniques provide better accurate indoor localization system [53][3][2]. The basic concept of this technique is to find the location of a mobile device by comparing its signal services and pattern received from the wireless communication transmitters like WiFi AP, and BLE [54][26][29][2]. The RADAR system was one of the examples of this typical technique [55][20]. Li *et al.* has presented indoor wireless localization using Bluetooth low energy beacons after the release of BLE protocol [26]. The traditional Bluetooth has a significantly long scan time, which limits its value for localization. WiFi-fingerprint-based approaches have received significant interests from industry and academia because of widespread deployment of access points (AP's) [35][50][51][36][29]. This approach is non-intrusive not requiring explicit user participation (although the operator needs to put much efforts to collect WiFi fingerprints); it does not require line-of-sight communication with AP's, and it has good localization accuracy in complex indoor environments [3][20].

3.3 WiFi-Fingerprint Based Indoor Localization

WiFi Fingerprinting-based indoor localization has become a better choice as it requires no extra infrastructure [1]. The fingerprint-based localization has a sequence of steps to follow [46]:

- **Fingerprint-Based Sensing:** Fingerprint sensing is the first step in any fingerprint based localization. When the system starts, the sensors are activated to record the data continuously. In our system, the RSSI (received signal strength) signatures are collected from the access points in the building.

- **Fingerprint Cleansing:** When receiving the RSS data from the access points of the building, the RSS data may contain noise which increases the chance of error. Due to this error, the accuracy of the actual location compared to the fingerprint-based signature or location will be low which degrades the performance of the whole system. To reduce this issue, a filtering process is required to clean the data before forwarding it to the next step.

- **Fingerprint Generation:** The fingerprint generation step takes the data after the filtering process and structure the data that can be easily parsed. There are different types of fingerprint data to construct the system. Out of which, we use

RSSI signatures in order to construct this system.

- **Fingerprint Matching:** In the fingerprint matching stage, the fingerprint signature of RSS collected and filtered by the fingerprint-based localization system has to be matched with the referenced fingerprints in the database. Based on the fingerprint match at that particular position the signature search space is assigned to that location. As the size of the reference points search space increases the matching performance of the system decreases. This particular issue was considered in our experimental setup with sampling period and sampling density (Sections 6.4 and 6.3). After creating the search space, a pattern matching algorithm is used in order to find out the device or person by comparing the current signature with the pre-recorded signature in the database.

- **Populating the Fingerprint Database:** This system needs a database of pre-recorded signatures or fingerprints for localization.

WiFi-fingerprint-based indoor localization is based on RSSI fingerprints [3][1][4]. Received signal strength indicator (RSSI) indicates the power in the received radio signal, and it was expressed with decibels (dB) with reference to one milliwatt (mW) with short-form dBm. The minimum signal strength for applications like WiFi that requires a reliable connection with the timely delivery of data is -67 dBm (RSSI value less than -67dBm is very impressive). WiFi-fingerprint based indoor localization schemes have two major phases (Figure 3.1).

- **Training Phase:** In the training phase, WiFi fingerprints (*i.e.,* received signal

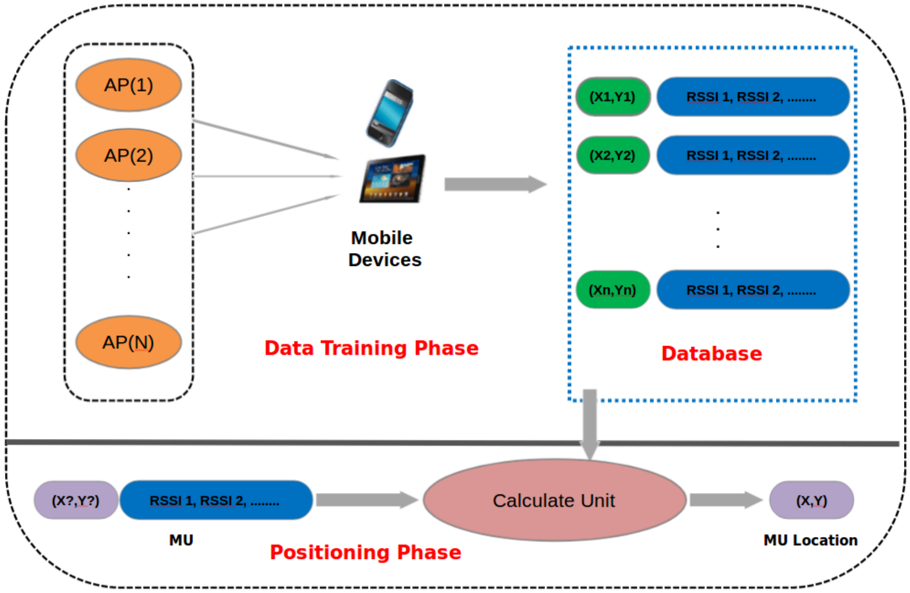

Figure 3.1: WiFi-Fingerprinting Based Position Estimation

strengths - RSS) of surrounding AP's are collected from each reference point of a target area. Obtained RSS data are accumulated in the RSS database to train and generate an RSS model for each reference point.

- **Testing Phase:** In the testing phase, the current location is determined by comparing measured RSS values with RSS models. Consequently, a reference point that has the closest match with measured RSS values is chosen as the current location.

3.4 UWB-Based Positioning System

Ultra-wide-band (UWB) wireless communication is an emerging technology in the field of indoor positioning which has better performance compared to others [22][20]. Due to the large bandwidth of UWB-based positioning systems, different frequency components show different interactions within the environment [31]. There are several UWB-based methods for the applications of positioning systems:

- **Signal-Based Methods:** Signal-based positioning methods calculates the distance between 2 nodes by measuring the energy of the received signal. The UWB has a high bandwidth which helps in calculating the distance between the transmitter and the receiver [31][20]. Due to the large bandwidth of UWB system, it has different frequencies in the environment. We need at least two nodes to make the estimation range. The first node is called as the tag, and the rest of the nodes are called as the anchors where each node has both transmitter and receiver. To find the range between the nodes, we need to know the characteristics of the channel of transmission. Based on the characteristics of the channel we can provide the accurate calibration which helps in position estimation accuracy. But there may be a chance in energy loss during the transmission of the signal from node to node with large distances which reduces the system accuracy. So, signal based approaches can be more accurate for only short distances [22][43][29].

- **Angle of Arrival-Based Methods**: Angle of Arrival(AoA)-based positioning methods are based on, the nodes sensing the direction of the signal [56][20]. Angle of arrival node sensing requires several antenna arrays or ultra-sound receivers [57][58]. After sensing the node signals at multiple antenna arrays, this signals are used to estimate the angle of arrival (AoA) based on the properties of the wireless channel from the node to the tag [58][22]. Antenna array can be defined as the natural extension of two to n antennas [56][59]. To estimate the location of the node, we need to measure the angles of the straight lines that connect the tag and the node. The position estimation in AoA-based approach can be determined with only three measuring units for 3D localization and two measuring units for 2D localization [43]. The inaccuracies of AoA-based approaches can be caused either by measuring device or the wireless communication channel [57]. The drawback of AoA-based approach is that, it needs to use a relatively large array of antennas which increases the cost of the system [60][43].

- **Time-Based Methods:** Time-based positioning methods are most promising due to their low cost hardware, and accurate range estimation [61][20]. The time-based methods, time of arrival (ToA) and time distance of arrival (TDoA) records signal propagation time from the transmitter to the receiver [62][47]. This can be explained by determining, the time of arrival (ToA) of the incoming signal from transmitter multiplied by the estimated ToA with the speed of light [47]. The UWB-ToA algorithms have better performance for indoor localization with tens of centimeters precision [63]. In our thesis, the time of arrival is calculated using UWB nodes. Each UWB-node has a common clock. The time taken by the signal propagation from the tag to the anchor is used to calculate the distance between the tag and the anchor [43][22]. If there is no synchronization between the tag and the node, then ToA-based approach can be deployed by using the synchronization between the node and other reference nodes in the field. In the absence of common clock with tag and the nodes, the round trip time of flight between either two nodes or tag and node can be used to make an estimation of the distance [22][63].

Among the above three approaches, the time-based method has received great attention for our system due to its accuracy compared to other approaches [63][47][43][20]. In particular, for our implementation of finding the position of a nano-scale unmanned aerial vehicle (NAV) in the field, we use two-way ranging [64]. The two-way ranging (TWR) method finds the distance between the tag and the anchor, by multiplying the estimated ToA of UWB-radio frequency signal with the speed of light [32][64].

In two-way ranging mode, the tag pings anchor with a POLL message to know the address of the anchor. The received time of POLL message from tag is recorded at anchor. The anchor replies with a response message to tag, and the tag records the round trip time of the message. Then the tag composes and sends the final message to anchor, and the time reception of final message report at tag is used to measure the distance between the tag and the anchor. The message sequence is shown in Figure 3.2.

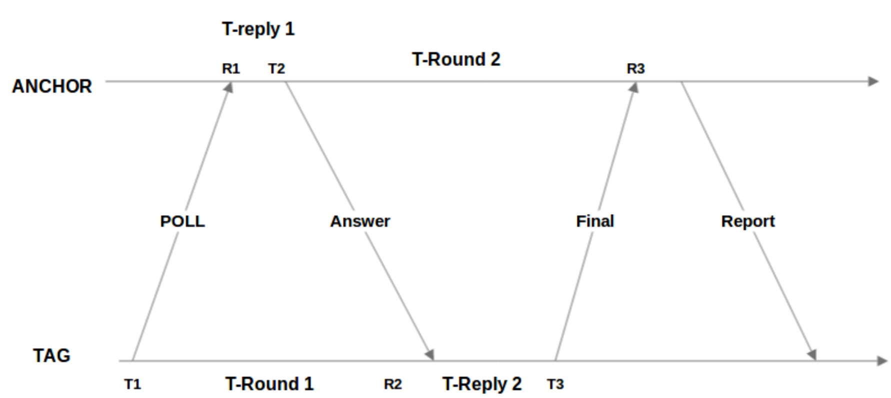

Figure 3.2: 2-Way Ranging Protocol

3.5 Unmanned Aerial Vehicles

Unmanned Aerial Vehicles (UAV) can be defined as an aerial vehicle which does not need a human operator. With the number of commercial drones and small-scale unmanned aerial vehicles has escalated very high demand in research towards various applications [65].

Figure 3.3: Crazyflie 2.0 (NAV)

Working with Nano-scale Unmanned Aerial Vehicles (NAV's) and Micro-scale Unmanned Aerial Vehicles (MAV's) are very advantageous like flying at low altitudes less than 100cm, ability to reach remote locations for monitoring purpose where a human find it hard to enter and also capture images with different angles, very lightweight and easy to carry to any place, the capacity of holding multiple sensors to its platform in order to acquire sensor data from the environment [66]. Based on all these advantages, the NAV's and MAV's are used for real-world applications like disaster management, industry monitoring, and inspection, precision agriculture, Aerial Imaging with high-quality films and videos.

Chapter 4

System Design

The proposed system consists of four main components: the quadcopter platform, indoor localization subsystem, semi-autonomous flight subsystem, and WiFi fingerprinting subsystem (Figure 4.1).

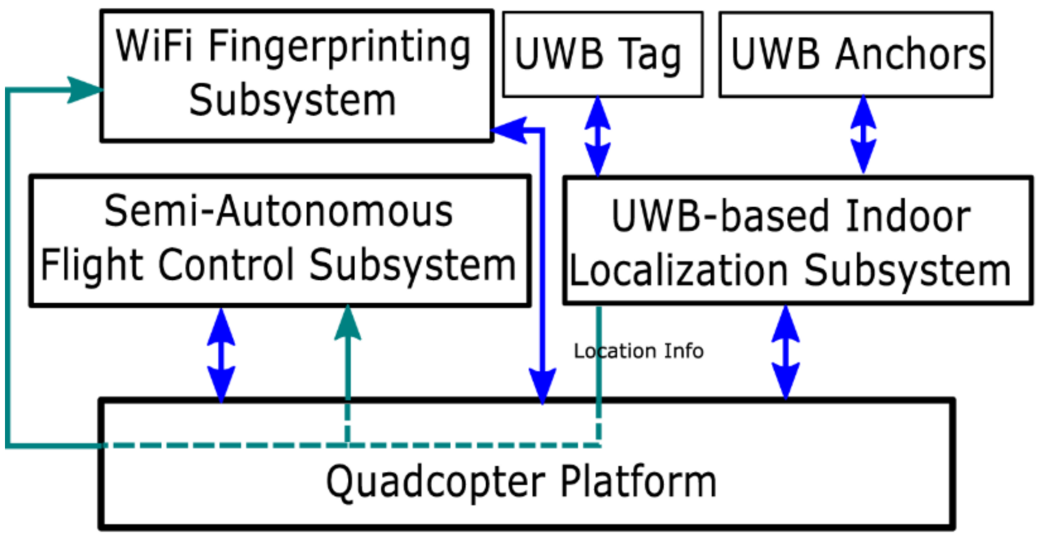

Figure 4.1: System Architecture

4.1 Quadcopter Platform

The quadcopter platform (Figure 4.1) supplies electric and computational power to the subsystems and provides interfaces to allow them to communicate with each other. The localization subsystem is primarily used to determine the current location of a NAV. This location information is used by the fingerprinting and the semi-autonomous flight control subsystems. More specifically, based on the current location, the semi-autonomous flight control subsystem manages the motion of a NAV, *i.e.,* it moves a NAV to the desired location and makes it hover to measure WiFi RSS using the WiFi fingerprinting subsystem. After collecting and analyzing RSS data, the WiFi fingerprinting subsystem sends the results to the RSS database server. The details of each subsystem are described in the following subsections.

We adopt the *Crazyflie 2.0* as our quadcopter platform [67] (Figure 4.2). It is a fully open-source platform that allows for easy integration of new hardware and software modules at a low cost. It is very small with dimensions of 9.2cm by 9.2cm, but it is equipped with powerful microcontrollers and numerous sensors including gyroscope, accelerometer, magnetometer, and barometer sensors allowing us to accurately control the motion. For this project, a micro WiFi module and a UWB-based indoor localization subsystem are integrated on this platform.

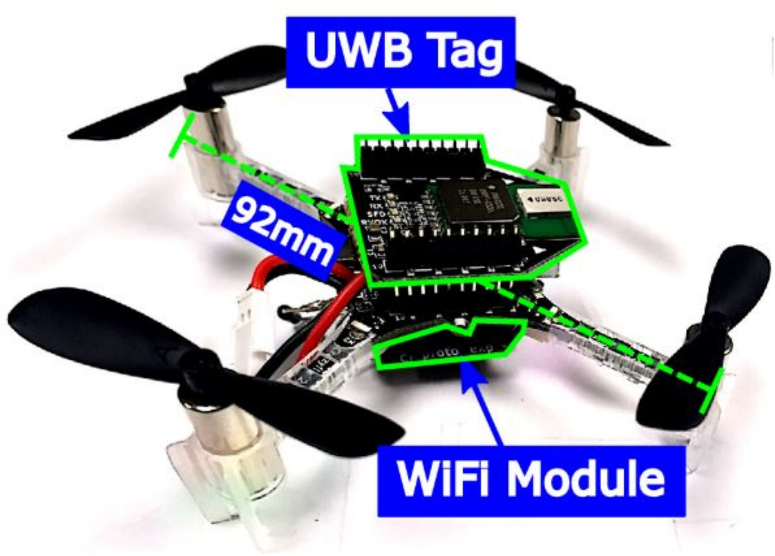

Figure 4.2: Quadcopter Platfom with Integrated UWB-Tag and WiFi Module

The quadcopter platform, however, has some limitations: its maximum payload and flight time are limited to 15g and 7 minutes, respectively. This payload issue for the proposed system is addressed by integrating a light-weight micro WiFi module. In addition, to address this flight-time issue, we propose to use *multiple NAVs* for WiFi fingerprinting, which is feasible due to the low price and small form factor of a NAV. More specifically, the spatio-temporal trajectories of multiple NAVs (*i.e.*, a spatially and temporally ordered sequence of set points) can be designed such that multiple reference points can be simultaneously covered.

4.2 NAV Localization

Ultra-wideband (UWB)-based indoor localization is a well-known technique for high localization accuracy [23]. There are numerous commercially available UWB-based localization systems. We adopt a commercially available UWB-based indoor localization system to implement the NAV localization subsystem [32]. It is worth to note that this UWB-based system cannot be directly used to localize off-the-shelf WiFi devices as it requires software and hardware modifications to those devices.

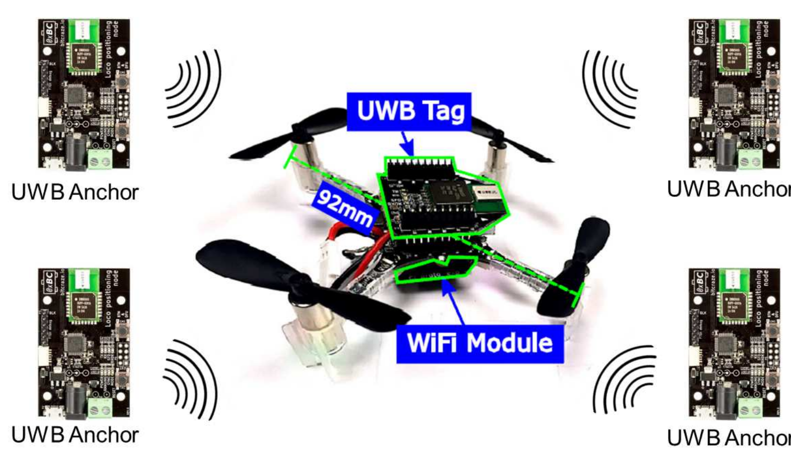

Figure 4.3: UWB-Based Localization Subsystem.

Using the time of arrival (ToA), distances between a NAV and anchors are measured (Figure 4.3). The Kalman filter-based position estimation approach is applied to estimate the location of a NAV based on these distance measurements [33]. Figure 4.4 depicts the locations of a hovering NAV at a pre-defined set point that we measured using the UWB-based indoor localization subsystem. The red circle in this graph indicates the set point and the blue circles indicate measured real-time NAV positions. The results show that the mean location error is approximately 9cm, which is reasonable to make a NAV hover and measure WiFi RSS data reliably.

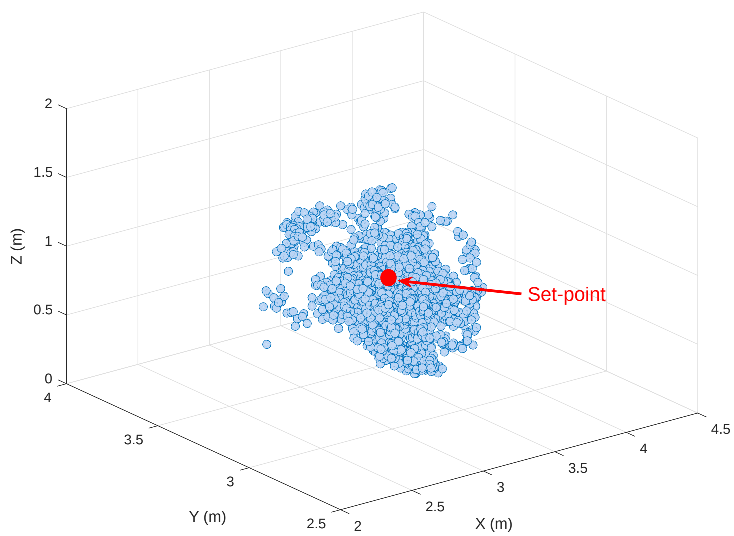

Figure 4.4: Locations of a hovering NAV measured using UWB-based localization subsystem.

4.3 Semi Autonomous Flight Control

The semi-autonomous flight control subsystem is used to control the motion of a NAV. The motion control algorithm of this subsystem is illustrated in Figure 4.5. More specifically, it consists of four individual PID controllers denoted by $PID_x, PID_y, PID_z, PID_{yaw}$ that control pitch, roll, thrust, and yaw to move a NAV to the desired set point denoted by $S_i = \{x_i, y_i, z_i\}$ defined on a 3D coordinate system. Given the current location of a NAV denoted by $\{x_c, y_c, z_c\}$, it is compared with the set point $\{x_i, y_i, z_i\}$, and the difference for each coordinate is calculated, *i.e.*, $|x_c - x_i|$, $|y_c - y_i|$, and $|z_c - z_i|$. If the difference is smaller than predefined thresholds denoted by T_x, T_y, and T_z for a predefined period of time, the NAV goes into the hover mode and initiates WiFi RSS measurements by communicating with the WiFi fingerprinting subsystem (Section 4.4). Once the predefined measurement period is expired, the pitch, roll, thrust, and yaw of the NAV is controlled to move to the next set-point, *i.e.*, S_{i+1}. Essentially, these set-points, as well as the hovering period, define the spatio-temporal trajectory of a NAV for WiFi fingerprint measurements.

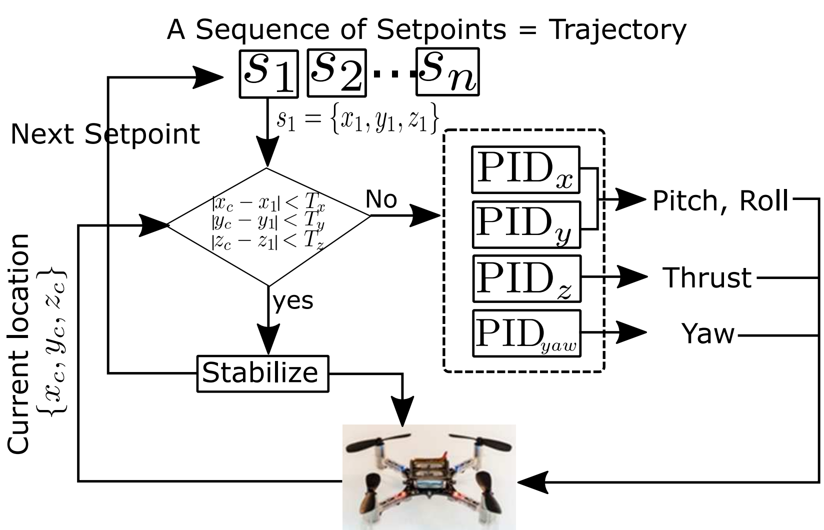

Figure 4.5: NAV Motion Control Algorithm.

4.4 WiFi Fingerprinting Subsystem

The WiFi fingerprinting subsystem collects WiFi RSS data when a NAV is in the hovering motion. To collect WiFi RSS data, we integrated the ESP8266 WiFi module. This module weights only 1.5g that is sufficiently light-weight to be integrated with the payload-limited quadcopter platform. It also draws a small amount of current of 60 to 220mA supporting the 802.11 b/g/n standard. However, potential challenges include interference with the UWB module and interference due to the propulsion of a NAV. We performed a microbenchmark to clarify these uncertainties.

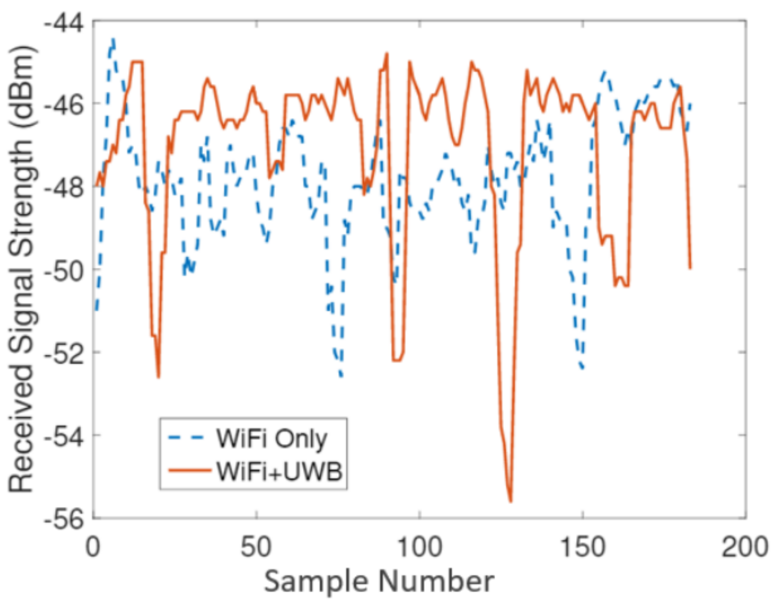

Figure 4.6: Interference between UWB and WiFi modules.

To test the potential interference between the UWB and WiFi modules, we measured the WiFi signal strength for WiFi-only and WiFi+UWB cases (*i.e.*, experimental and control groups respectively). Figure 4.6 displays the results. It is shown that the mean signal strength for the WiFi+UWB case was -47.8dBm and the signal strength for the WiFi-only case was -47.0dBm, which confirms that there is no noticeable effect of the UWB signal on WiFi RSS measurements.

Furthermore, to evaluate the effect of the NAV propulsion on signal strength, we varied the propeller rotation rates and measured the received WiFi signal strength. Figure 4.7 displays the mean signal strengths of RSSI. As the results demonstrate, there is little effect of the propulsion of a NAV on WiFi fingerprinting.

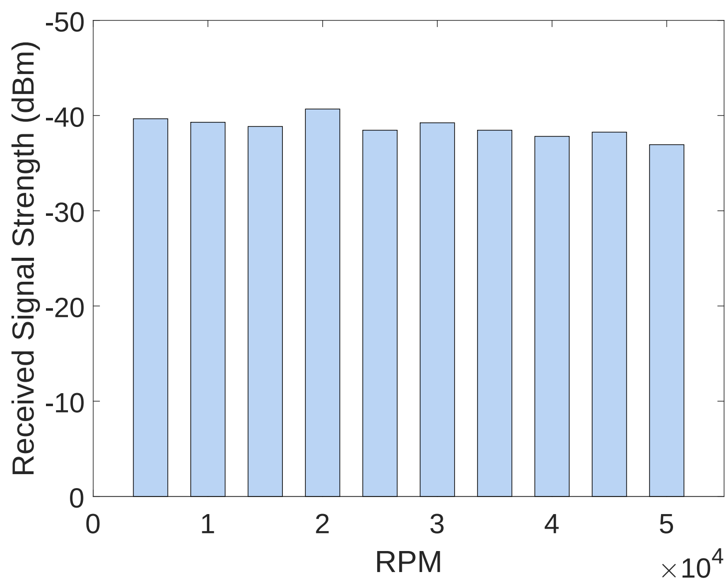

Figure 4.7: Interference due to NAV Propulsion.

The minimum signal strength for applications like WiFi that requires a reliable connection with the timely delivery of data is -67 dBm and RSSI value less than -67dBm is very impressive.

Chapter 5

System Implementation

In our approach of automating WiFi-Fingerprinting using nano-scale unmanned aerial vehicle for indoor localization, we integrated several subsystems to Crazyflie 2.0 (NAV) to design the complete system architecture. Before the integration of system design, we worked with each subsystem environment to know how they function, what are the communication channels of the subsystems, what kind of result that each of the subsystems provides, and learned how to use those subsystems to design a complete system architecture. In this chapter, we explain the background information of each subsystem.

5.1 WiFi Fingerprinting Subsystem

WiFi fingerprinting in our proposed system plays a key role in collecting RSSI signatures to provide accurate indoor localization system. For this particular system, we used an ESP8266 WiFi module, which is a low-cost microchip with an on-board micro-controller and a full stack TCP/IP protocol. The features of the ESP8266 chip are as follows [68]:

- Integrated TCP/IP protocol stack.
- WiFi Direct (P2P), Soft-AP, and supports 802.11 b/g/n protocol.
- Standby power consumption of less than 1.0 mW.
- Wake up and transmit packets in less than 2ms.

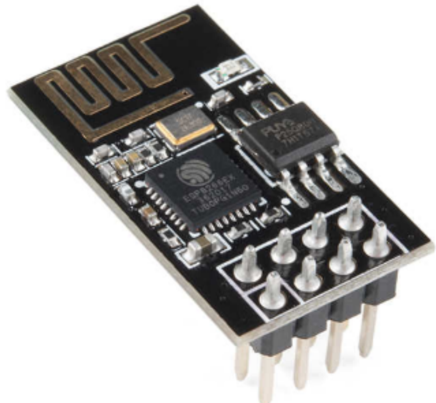

Figure 5.1: ESP8266 WiFi Module

- On-chip 1MB Flash and can support up to 16 Mb.
- Supports antenna diversity.
- Integrated temperature sensor.
- Integrated low-power 32-bit CPU could be used as an application processor.
- Integrated PLL, regulators, and power management units.
- Weight is less than 1.5 g.

The ESP8266 WiFi module was initially integrated to Arduino-Uno to learn how it works and at what baud-rate the ESP8266 module communicates with the system. There are many internet resources on integration of ESP8266 WiFi module with Arduino as this module is used in several industrial applications [69]. The connections between the Arduino module and the WiFi module are shown in the Figure 5.2:

The connection is as follows:

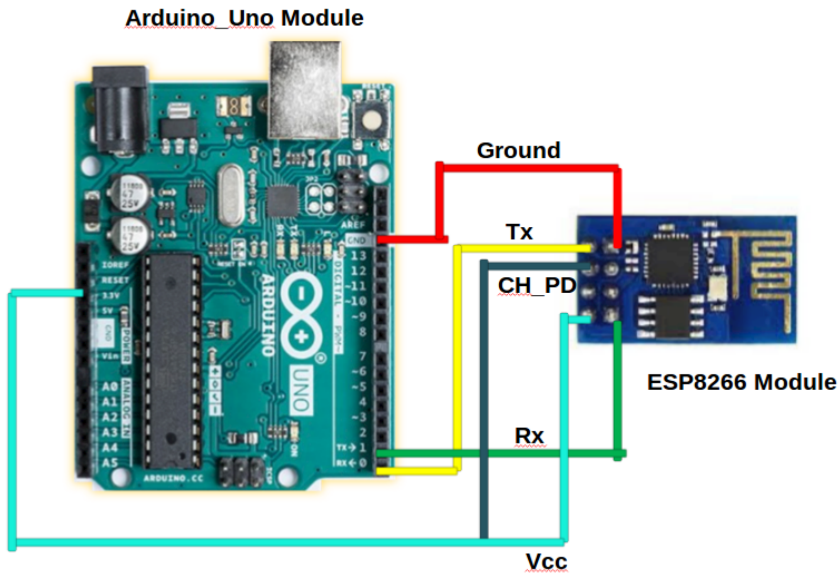

Figure 5.2: ESP8266 Connection with Arduino-Uno Module

- The reset pin is connected to the Arduino-UNO ground to bypass the Arduino boot-loader.

- The VCC of the ESP8266 module is connected to VCC of the Arduino module. Here we need to consider the power capacity of the ESP8266 module. The ESP8266 module needs only 3.3V of power where the supplying more voltage may damage the module. So for reducing the voltage to ESP8266, we use a diode.

- The transmitter Tx of ESP8266 module is connected to the receiver Rx of Arduino module. Moreover, the receiver Rx of ESP8266 module is connected to transmitter Tx of the Arduino module.

- CH_PD is a chip controlled power down pin which is connected to the power supply VCC of Arduino-Uno.

- The GPIO input/output pins are connected to the GPIO's of the Arduino.

After connecting ESP8266 WiFi module to Arduino-UNO, we need to set the target board in Arduino-IDE to 'Arduino Uno' (Tools/Board/Arduino-Uno). Then we can use the ESP8266 WiFi module by adjusting the serial monitor of Arduino-IDE with 'both NL and CR' and the 'baud-rate'. The ESP8266 WiFi module in our system communicated at '115200' baud-rate. But some WiFi modules communicates at different baud-rates like '9600', '57600', *etc* [69]. We used 'AT' command to communicate with ESP8266 WiFi module. The 'AT' command is an instruction used to control the modem. AT is an abbreviation of ATtension. After sending the 'AT' command, the WiFi module acknowledges with an 'OK' message. If not, then we have to check with different baud-rate.

```
+CWLAP:(3,"CVBJB",-71,"f8:e4:fb:5b:a9:5a",1)
+CWLAP:(3,"HT_00d02d638ac3",-90,"04:f0:21:0f:1f:61",1)
+CWLAP:(3,"CLDRM",-69,"22:c9:d0:1a:f6:54",1)
+CWLAP:(2,"AllSaints",-88,"c4:01:7c:3b:08:48",1)
+CWLAP:(0,"AllSaints-Guest",-83,"c4:01:7c:7b:08:48",1)
+CWLAP:(0,"AllSaints-Guest",-83,"c4:01:7c:7b:05:08",6)
+CWLAP:(4,"C7FU24",-27,"e8:94:f6:90:f9:d7",6)
+CWLAP:(2,"AllSaints",-82,"c4:01:7c:3b:05:08",6)
+CWLAP:(3,"QGJTL",-87,"f8:e4:fb:b5:6b:b4",6)
+CWLAP:(4,"50EFA8",-78,"74:44:01:50:ef:a7",6)
+CWLAP:(0,"optimumwifi",-78,"76:44:01:50:ef:a8",6)
+CWLAP:(3,"BHQH4",-95,"18:1b:eb:1a:af:5b",6)
+CWLAP:(3,"NETGEAR49",-86,"84:1b:5e:e0:28:03",7)
+CWLAP:(3,"ngHub_319332NW00047",-56,"20:e5:2a:79:b1:2f",11)
+CWLAP:(3,"BFZR4",-73,"18:1b:eb:1d:c3:91",11)
+CWLAP:(1,"5FFVL",-82,"00:26:b8:b5:c0:f2",11)
+CWLAP:(3,"59G6D",-77,"00:7f:28:6d:91:7b",11)
+CWLAP:(3,"N16FU",-53,"20:cf:30:ce:60:fe",11)
+CWLAP:(3,"ITS",-82,"90:72:40:21:5f:76",11)
+CWLAP:(3,"ITS",-79,"24:a2:e1:f0:04:e4",11)
```

Figure 5.3: ESP8266 WiFi Module Response for AT+CWLAP Command

In our system, we need RSSI signatures for indoor localization which is obtained by sending 'AT+CWLAP' as a command to ESP8266 WiFi module. The

'AT+CWLAP' command gives all the available access points with a response message as shown in Figure 5.3. The message format of 'AT+CWLAP' command has ECN (explicit congestion notification), SSID (String, SSID of AP), RSSI (signal strength), MAC (string, MAC address) as parameters. After complete understanding of ESP8266 WiFi module, we integrated it to Crazyflie 2.0 subsystem for collecting WiFi RSSI data. The RSSI data collected by the ESP8266 WiFi module is sent to the server system through a 'Crazyflie 2.0 subsystem' and 'RoS subsystem'. The parameters of AT+CWLAP command response is used to construct RSSI fingerprint database. The RSSI (signal strengths) with their respective MAC address and the list of all AP's in the communication range are used as contents to calculate the mean signal strength of collected RSSI signatures at every set-point in the experimental field. At every set-point, the RSSI is measured for 2 minutes to set up the database. This is done at all the equally spaced set-points in the experimental field. The RSSI measurement for every 2 minutes has provided with 200 samples approximately. Next, we performed localization at each set-point and measure the results of the localization. And this step is repeated for ten times at every set-point. The results of the localization are given as coordinates in the fingerprint database. Sample coordinates provided to fingerprint database for localization is as follows: P1 {(a.b),(a1,b1),........,(a10,b10)}, P2 {(a.b),(a1,b1),........,(a10,b10)}, P3 {(a.b),(a1,b1),........,(a10,b10)}, P4 {(a.b),(a1,b1),........,(a10,b10)}, ..., P30 {(a.b),(a1,b1),........,(a10,b10)} where P {1,2,3,4,.....,30} are equally spaced set-points and {(a.b),(a1,b1),........,(a10,b10)} are localization results at each set-point in the field. Then the new RSSI signatures are compared with the existing RSSI signature database for indoor localization.

5.2 Nano Scale Unmanned Aerial Vehicle

Nano-scale unmanned aerial vehicles have more advantages to work in an indoor environment such as stability, can take-off and land vertically, and easy to control. So, we have chosen Crazyflie 2.0 as our nano-scale unmanned aerial vehicle in our system design. Crazyflie 2.0 is an open-source lightweight (29g) nano quad-copter NAV which makes it ideal for our approach [67].

Figure 5.4: Crazyflie 2.0 - NAV

The features of Crazyflie 2.0 are as follows [70]:

- STM32F405: Main micro-controller (with Cortex-M4, 168 MHz, 192 kB SRAM, 1MB flash), used for state-estimation, control, and handling of extensions.
- nRF51822: radio and power management micro-controller (with Cortex-M0, 32

MHz, 16 kB SRAM, 128 kB flash). We call this nRF51.

- MPU-9250: 9-axis inertial measurement unit.
- Several onboard sensors including gyroscope, accelerometer, magnetometer, and barometer sensor which helps in stabilizing the altitude of the NAV.
- LPS25H: pressure sensor.
- 8 kB EEPROM.
- uUSB: charging and wired communication.
- Expansion ports with UART, GPIO.
- Debug port for STM32.

A Crazyflie can communicate with both smartphone and a PC as well. The Crazyflie is controlled by a game controller and a custom USB dongle called Crazyradio (Shown in Figure 5.5). Crazyradio PA (PA - power amplifier) is a long-range open USB dongle based on the nRF24LU1+ Nordic-semiconductor. The power amplifier (PA) of Crazyradio boosts the communication range of up to 1km from Crazyflie to Crazyradio PA and 2km range from Crazyradio PA to Crazyradio PA.

To configure the Crazyradio and Crazyflie, we used Ubuntu 14.04 as an operating system by downloading the latest Crazyflie PC client installer [67]. After installing the PC client, we configured the Crazyradio channel to communicate with Crazyflie. By default, the Crazyflie radio has a communication channel of '**radio://0/80/250K**'.

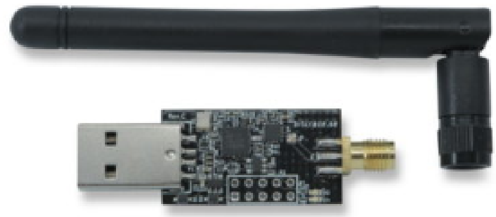

Figure 5.5: Crazyradio with PA

After successful connection of Crazyflie to Crazyflie-client through Crazyradio, we tested the Crazyflie connection by hovering it in indoor environment. Then we integrated the WiFi-Fingerprinting subsystem to the Crazyflie 2.0 NAV and hovered it in an indoor environment to collect the RSSI signatures for indoor localization.

5.3 Robotic Operating System (ROS)

As Nano-scale unmanned aerial vehicle system is considered under the stream of robotics, ROS provides frameworks to work with robotic middle-ware applications. ROS is a collection of hardware drivers, tools, and algorithms for building robotic applications [71]. It is primarily used to construct the structure of a functioning software for robotic applications. ROS visualization package helps the Crazyflie with UWB positioning environment in a 3-dimensional view. So, we used ROS for controlling Crazyflie 2.0. To configure ROS, we have chosen 'ROS-Kinetic' as our platform which is compatible with our Ubuntu system version [67][70]. To control the Crazyflie, we need to configure 'Ros-kinetic-joy' which supports the libraries of the game controller.

To clone the required repositories of Crazyflie 2.0, we need to create a 'catkin_workspace' [32]. After creating the workspace, we need to source the workspace with '**source devel/setup.bash**' command and set up the 'package_path' environment variable directory to compile the catkin workspace. Inside the catkin workspace, we need to clone the 'ROS-loco positioning system directory' and 'crazyflie ROS directory' and build those packages with the help of 'catkin_make' command [32]. The Crazyflie hovers with a launch command which has Crazyflie loco-positioning system estimator, UWB-based location estimator with extended Kalman filter (ekf), Crazyflie communication radio channel and the user given coordinate position in XYZ-plane as parameters of the command.

Command: roslaunch bitcraze_lps_estimator dwm_loc_ekf_hover.launch uri:=radio://0/30/250K x:=2.0 y:=0.5 z:=1.5

Based on the XYZ coordinates given by the user the Crazyflie hovers to that position and start collecting the RSSI signatures and send them to the server system.

5.4 UWB-Based Indoor Positioning

The UWB based indoor positioning system in our system design, uses Crazyflie 2.0 expansion deck with UWB-tag and anchor setup which helps in finding the position of the NAV in an indoor environment [32][67]. The UWB module is attached to both the tag (Crazyflie 2.0 expansion deck) and the anchor setup for accurate position estimation. The UWB-based indoor positioning subsystem consists of six anchors which are mentioned as references to find the position of the tag based on the Time of Arrival (ToA) of the signal from anchor to anchor and anchor to tag [67].

- UWB-Tag: The UWB-tag is attached as an expansion deck to the Crazyflie while hovering. There is no need to configure UWB-Tag externally. The firmware of UWB-tag activates directly after updating the latest firmware of Crazyflie 2.0. The tag is considered by default as the base tag with starting value as 0, and the rest of the referenced anchors are counted from value 1 to rest in the experimental field.

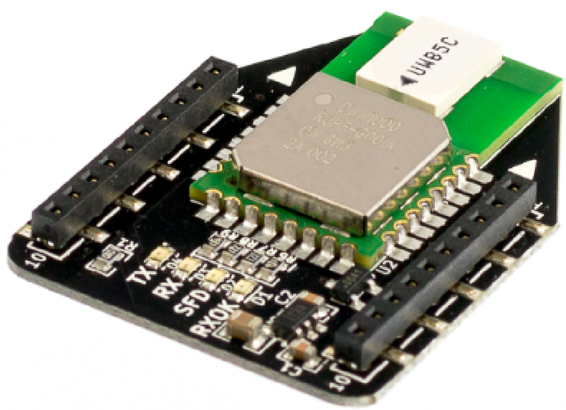

Figure 5.6: UWB - Tag

The tag contains a row of 4 LED's that blink:

  - **TX**: Transmitting.
  - **RX**: Receiving.
  - **SFD**: Received Packet Time-Stamp.

– **RXOK**: Packet Received without error.

The distance from anchor to tag and anchor to anchor is used to estimate the absolute position of the Crazyflie in the experimental setup. The location of the node is calculated on Crazyflie itself, and there is no need of external configuration for position estimation. Based on the estimated position of the tag we can autonomously hover the NAV to the desired location to collect the WiFi RSSI signatures.

- UWB-Node: The UWB-node is a versatile positioning device that can run either as a reference node or a tag in the environment setup [32].

  The node can be used as:

  – Anchor: which receives and answers ranging request.

  – Tag: which ranges with anchors and prints the distance.

  – UWB-sniffer: which prints all the messages from the radio

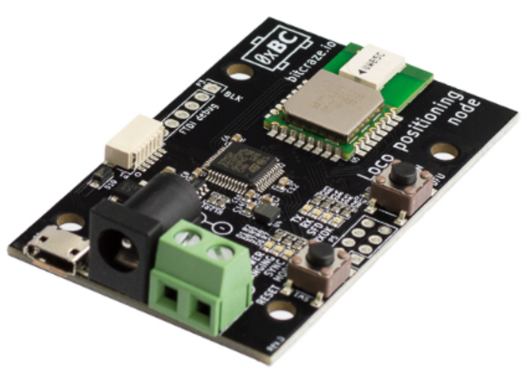

Figure 5.7: UWB- Anchor (or) Reference

We need to update the latest firmware to each anchor (UWB-node) before deploying them in the experimental field. To configure the node we need to connect it to a PC running the Crazyflie-client. The node is recognized as a serial port and the serial port is found with the help of running **'dmesg'** command in the Crazyflie-client terminal [32]. This command opens a console window, and then we need to configure the node with anchor mode and the number of that particular anchor.

```yaml
n_anchors: 6
anchor0_pos: [  0,   0, 1.85]
anchor1_pos: [  0,   2, 1.85]
anchor2_pos: [  4,   2, 1.85]
anchor3_pos: [  4,   0, 1.85]
anchor4_pos: [1.3,   1, 1.85]
anchor5_pos: [2.6,   1, 1.85]
```

Figure 5.8: Anchors Positions in XYZ-Plane

The anchor number and position of the anchor in XYZ-plane should be updated to file **'anchor-position.yaml'** (Shown in Figure 5.8) in 'ROS-loco positioning system directory' before launching the 'roslaunch' command [67].

Example Command: roslaunch bitcraze_lps_estimator dwm_loc_ekf_hover.launch uri:=radio://0/30/250K x:=2.0 y:=0.5 z:=1.5

The performance of the positioning system will increase by placing more number of anchors in the field. But the cost of the system will gradually increase. The anchor setup will affect the system if the UWB-anchors are placed with a distance more then their communication range. But within their communication range, there is no much impact of the anchor set-up as the UWB node and the anchor setup does not need a clear line of sight communication. The performance of the positioning system can be reduced even with erroneous in the distance measurements of the anchors in the experimental field.

5.5 Work-Environment

After complete understanding of each subsystem, we integrated ESP8266 WiFi module and UWB-deck to the Crazyflie for complete system design. Then we configured the reference nodes by their positions, and the XYZ values of the anchors are given as an input to the ROS system as explained in section 5.4. To configure the anchors, battery powered backup devices are used to deploy the anchors with the help of a stand set-up to hold the anchors at desired heights in the experimental field. Initially, the NAV was placed on the floor and hovers to the set-point with user-given XYZ-coordinates, when the following command runs in the catkin_workspace:

Example Command: roslaunch bitcraze_lps_estimator dwm_loc_ekf_hover.launch uri:=radio://0/30/250K x:=2.0 y:=0.5 z:=1.5

So based on the above command, the ROS environment will launch:

- bitcraze_lps_estimator: which provides the environment of UWB based loco positioning system [32][70].
- dwm_loc_ekf_hover.launch: which launches the Crazyflie to take-off from the ground to the user provided XYZ-coordinate position in the experimental field

[32][70].

Figure 5.9: 3D visualization of Positioning NAV with UWB-Anchor Setup

The radio channel 'uri:=radio://0/30/250K' in this command uses 30Hz of frequency with 250K bits/sec to transmit data from the Crazyradio PA to the Crazyflie in the field. After Launching the command, the Crazyflie takes-off from the ground and hovers at the given XYZ-coordinates in the experimental field. The experimental setup with the positioned anchors and hovering Crazyflie is shown in a 3D environment in the server system using the visualization packages **'.rviz'** provided by ROS. The 3D visualization of the experimental setup is shown in Figure 5.9. In the above 3D visualization, the red color boxes indicate the anchors placed at the given XYZ-positions and the green box indicates the position of the Crazyflie carrying the UWB-tag and ESP8266 WiFi module to collect RSSI signatures at user-specified set-points. And the rest of the implementation of our system is explained precisely in section 4.

Chapter 6

Experimental Results

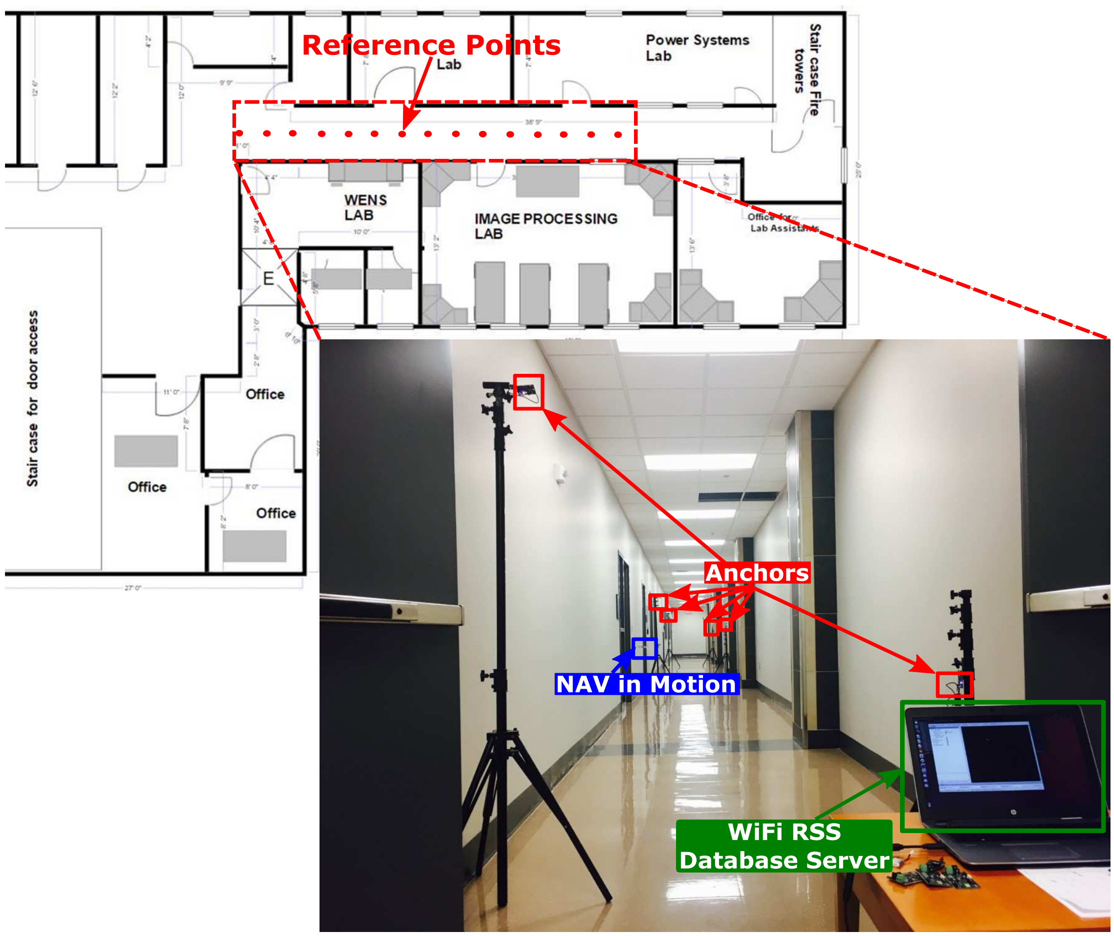

Figure 6.1: Experimental Setup.

Experiments were performed in the hallway of a department building. Six anchors were deployed to cover the hallway with dimensions of 30m x 2m. The reference points were defined at a 2-meter interval in the middle of the hallway (Figure 6.1). For this set of experiments, we used multiple fully charged NAVs to cover all the reference points. More specifically, another NAV was injected once a NAV returns before it depletes its battery. when the site surveying is performed, there is no obstacle present in the experimental setup. We leave the development of a more sophisticated trajectory planning algorithm as our future work.

An experiment to evaluate the packet delivery rates was conducted first to ensure that obtained WiFi RSS data would be reliably delivered to the RSS database server. We then measured the localization accuracy by varying the sampling period and the sampling density. The sampling period indicates the amount of time a NAV hovers to collect WiFi RSS data at one reference point, and the sampling density means the total number of reference points. The open-source *'Find'* WiFi fingerprint-based indoor localization solution was adopted to run WiFi-fingerprint based localization [72].

6.1 Packet Delivery Rates (PDR)

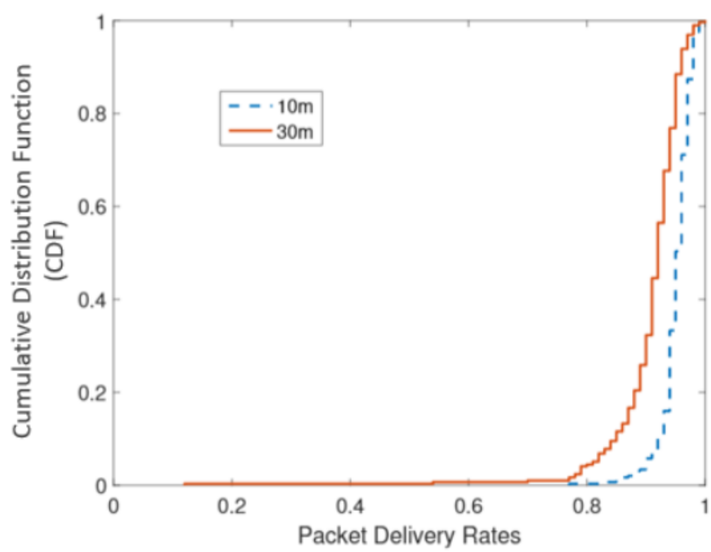

Figure 6.2: Packet Delivery Rates

The packet delivery rates, *i.e.*, $\frac{\text{num packet received}}{\text{num packet sent}}$ is a critical factor that affects the accuracy of indoor localization of the proposed system. We concentrate on verifying that collected WiFi RSS data are reliably received at the RSS database server. For this experiment, we made a NAV hover at different locations and measured packet delivery rates for transmitted packets containing WiFi RSS data while NAVs are hovering. Results are depicted in Figure 6.2. As shown, high packet delivery rates were achieved, *i.e.,* an average of 95% and 92% for NAVs hovering at 10m and 30m away from the RSS database server, respectively.

6.2 Effect of 3D Fingerprinting

One of the major benefits of using NAVs to perform WiFi fingerprinting is that we can easily obtain '3D WiFi RSS data'. To see the effect of the 3D WiFi fingerprinting, we performed WiFi fingerprinting at different heights, *e.g.,* at 50cm and 150cm. We then measured the localization accuracy with and without the WiFi RSS data measured at the height of 150cm.

First, we performed localization 30 times at each reference point based on the RSS database built only with the WiFi RSS measurements at the height of 50cm. We calculated an average localization error at each reference point. We then configured the database by adding new RSS measurements obtained at the height of 150cm. Interestingly, the results show that when the WiFi RSS data obtained at the height of 150cm were included, the location errors measured at the height of 50cm remarkably increased (Figure 6.3). These results indicate that for the better localization accuracy, separate reference points must be defined at different heights and the RSS database for these reference points need to be trained with separate WiFi RSS data obtained from different heights. The proposed system facilitates this process of building the WiFi RSS database with 3D WiFi RSS data.

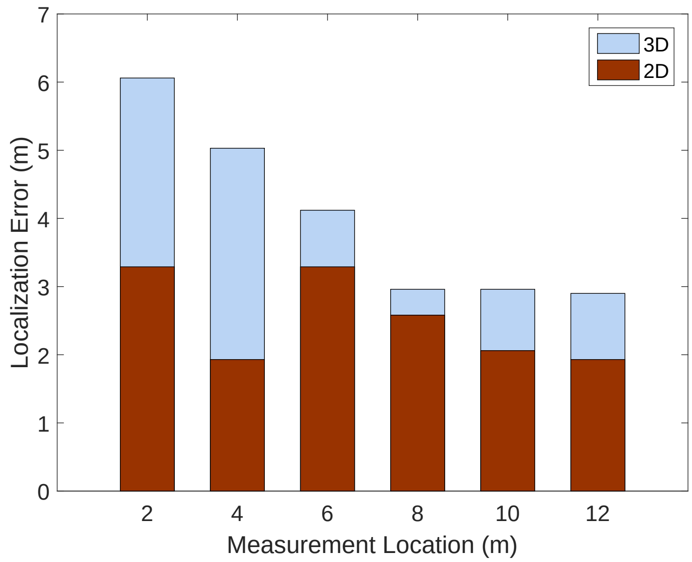

Figure 6.3: Effect of 3D Fingerprinting

6.3 Effect of Sampling Period

An experiment was performed to see the effect of the sampling period. More specifically, we performed localization at each reference point for five times and calculated the average location error for different sampling periods of 30sec, 60sec, and 90sec. The results are depicted in Figure 6.4. It was observed that as we increased the sampling period, we obtained better localization accuracy. Overall, the average localization errors were 2.46m, 1.79m, and 1.7m, for the sampling periods of 30sec, 60sec, and 90sec, respectively. It is also worth to mention that this relatively high localization accuracy is attributed to many WiFi APs in the department building.

Another important experiment is to compare the results with the manual WiFi

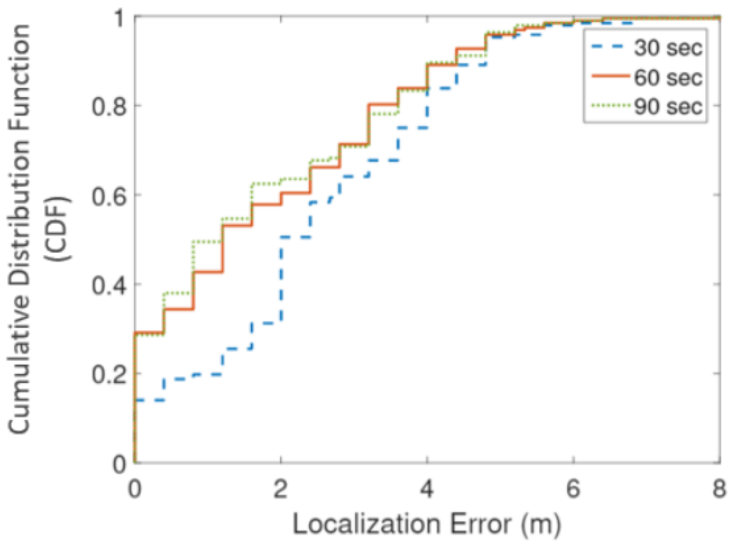

Figure 6.4: Effect of Sampling Period - Automated

RSS survey method, *i.e.*, collecting WiFi RSS data by holding a NAV at each reference point. So we repeated the same experiment for the manual WiFi RSSI fingerprinting. The results are displayed in Figure 6.5.

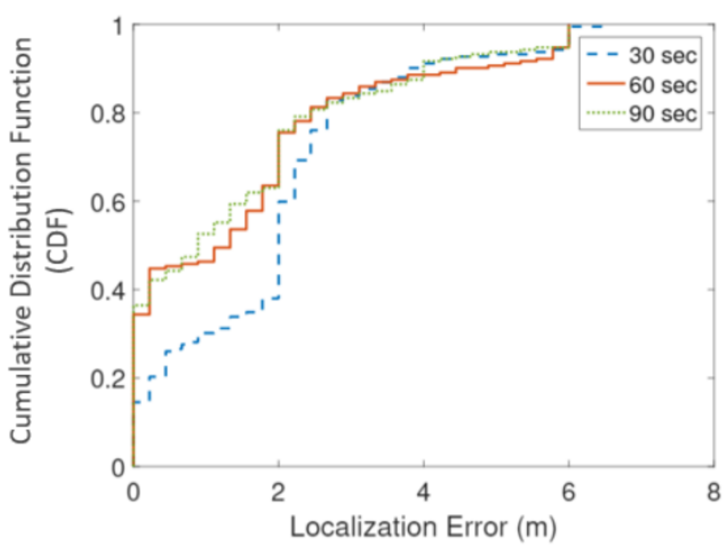

Figure 6.5: Effect of Sampling Period - Manual

The average localization errors for the manual WiFi fingerprinting were 1.97m, 1.54m, and 1.45m for the sampling periods of 30sec, 60sec, and 90sec, respectively, which is about 19%, 13%, and 14% decrease in the localization error compared with the NAV-based method. This performance gap is primarily attributed to the fact that a NAV continuously adjusts its position to hover at a reference point, while the manual fingerprinting method assures that a NAV is fixed at a reference point.

6.4 Effect of Sampling Density

In this section, the effect of the sampling density, *i.e.,* the total number of reference points is investigated. Similar to the experiment in Section 6.3, we performed localization at each reference point for 5 times and calculated the average localization error. This time, however, we varied the sampling density, *i.e.,* we used a different interval between reference points–1m instead of 2m. Results are depicted in Figure 6.6.

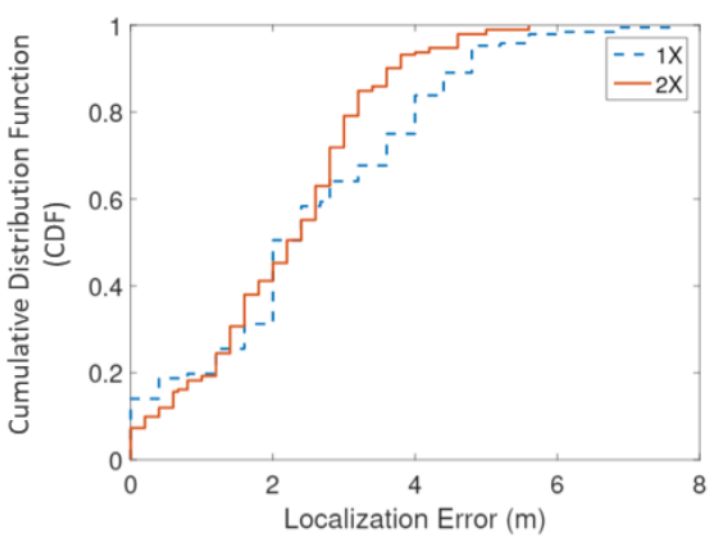

Figure 6.6: Effect of Sampling Density

It was observed that the increased density resulted in reduced localization errors. Overall, the average localization errors were 2.5m, and 2.2m for the low and high densities, respectively. We can also perform with sampling densities less than 1m like up to 0.10m for a more accurate system. But, when collecting the WiFi fingerprints with NAV subsystem, the mean localization error of NAV to hover at any particular set-point is 0.09m which is reasonable. But with the sampling density of 1m interval gap, we have achieved the system with localization error that is close to the manual fingerprint system by reducing the time and effort in collecting the RSSI signatures.

Chapter 7

Conclusion

In this thesis, a new trend of NAV-based indoor Wi-Fi fingerprinting survey system has been designed and implemented successfully which not only reduced the time and effort for the manual gathering of Wi-Fi-fingerprints but also presented the potential of increased indoor localization accuracy by collecting 3D fingerprint information in comparison with contemporary 2D Wi-Fi fingerprint collection methods. Also, the experimental results of the proposed system have demonstrated that automated NAV-based Wi-Fi fingerprints are effectively collected resulting in competitive indoor localization accuracy compared to the manual Wi-Fi fingerprint survey approaches.

Chapter 8

Future Research Directions

This thesis warrants a number of interesting future research directions. A critical issue of the proposed system is the limited flight time of NAVs. We addressed this issue by using a group of UAVs that take turns. However, different approaches can be applied. For example, extremely efficient advanced solar cell technologies, specifically designed for UAVs, have been developed [73]. We expect that in the near future, continuous operation of NAVs will be possible with these kinds of advanced solar cells extending the application domains of NAVs to military surveillance, law enforcement, fire fighter missions, disaster response, and so on. Another potential technology to address this challenge is to employ inductive charging systems that wirelessly recharge the batteries of NAVs. Another related future work is to design an effective multi-NAV coordination algorithm to minimize the WiFi RSS survey time, and development and employment of inductive charging stations that wirelessly charge NAVs.

Bibliography

[1] Jianwei Niu, Bowei Wang, Lei Shu, Trung Q Duong, and Yuanfang Chen. Zil: An energy-efficient indoor localization system using zigbee radio to detect wifi fingerprints. *IEEE Journal on Selected Areas in Communications*, 33(7):1431–1442, 2015.

[2] Ali Yassin, Youssef Nasser, Mariette Awad, Ahmed Al-Dubai, Ran Liu, Chau Yuen, and Ronald Raulefs. Recent advances in indoor localization: A survey on theoretical approaches and applications. *IEEE Communications Surveys & Tutorials*, 2016.

[3] Hong Li, Limin Sun, Haojin Zhu, Xiang Lu, and Xiuzhen Cheng. Achieving privacy preservation in wifi fingerprint-based localization. IEEE Conference on Computer Communications on IEEE INFOCOM - 2014.

[4] Suining He and S-H Gary Chan. Wi-fi fingerprint-based indoor positioning: Recent advances and comparisons. *IEEE Communications Surveys & Tutorials*, 18(1):466–490, 2016.

[5] Jun-geun Park, Ben Charrow, Dorothy Curtis, Jonathan Battat, Einat Minkov, Jamey Hicks, Seth Teller, and Jonathan Ledlie. Growing an organic indoor location system. In *Proc. of MobiSys*, 2010.

[6] Jonathan Ledlie, Jun-geun Park, Dorothy Curtis, André Cavalcante, Leonardo Camara, Afonso Costa, and Robson Vieira. Molé: a scalable, user-generated wifi positioning engine. *Journal of Location Based Services*, 6(2):55–80, 2012.

[7] Chenshu Wu, Zheng Yang, Yunhao Liu, and Wei Xi. Will: Wireless indoor localization without site survey. *IEEE Transactions on Parallel and Distributed Systems*, 24(4):839–848, 2013.

[8] Nam Tuan Nguyen, Rong Zheng, and Zhu Han. UMLI: An unsupervised mobile locations extraction approach with incomplete data. In *Proc. of WCNC*, 2013.

[9] Hua-Yan Wang, Vincent W Zheng, Junhui Zhao, and Qiang Yang. Indoor localization in multi-floor environments with reduced effort. In *Proc. of Percom*, 2010.

[10] Sameh Sorour, Yves Lostanlen, Shahrokh Valaee, and Khaqan Majeed. Joint indoor localization and radio map construction with limited deployment load. *IEEE Transactions on Mobile Computing*, 14(5):1031–1043, 2015.

[11] Myounggyu Won, Ashutosh Mishra, and Sang H Son. Hybridbaro: Mining driving routes using barometer sensor of smartphone. *IEEE Sensors Journal*, 17(19):6397–6408, 2017.

[12] Myounggyu Won, Haitham Alsaadan, and Yongsoon Eun. Adaptive audio classification for smartphone in noisy car environment. In *Proceedings of the 2017 ACM on Multimedia Conference*, pages 1672–1679. ACM, 2017.

[13] Aawesh Shrestha and Myounggyu Won. Deepwalking: Enabling smartphone-based walking speed estimation using deep learning. *arXiv preprint arXiv:1805.03368*, 2018.

[14] Myounggyu Won, Yunfan Zhang, XiaoZhu Jin, and Yongsoon Eun. Wiparkfind: Finding empty parking slots using wifi. In *2018 IEEE International Conference on Communications (ICC)*, pages 1–6. IEEE, 2018.

[15] Why doesn't gps work inside a building? `https://itstillworks.com/doesnt-gps-work-inside-building-18659.html`. Accessed: 2018-06-09.

[16] Torben Godsk Thomas T Dan Lund Christensen Kaj G Mikkel BK, H Blunck. Indoor positioning using gps revisited. *Pervasive Computing: Lecture Notes in Computer Science*, 6030, 2010.

[17] Muller H Randall C. Low cost indoor positioning system. *Ubiquitous Computing: Lecture Notes in Computer Science*, 2201, OCT - 2001.

[18] A positioning system that goes where gps can't. `https://www.scientificamerican.com/article/indoor-positioning-system/`. Accessed: 2018-06-09.

[19] Sudarshan S. Chawathe. Low-latency indoor localization using bluetooth beacons. *2009 12th International IEEE Conference on Intelligent Transportation Systems*, pages 1–7, 2009.

[20] Mautz Rainer. Indoor positioning technologies. *Ph.D. Thesis*, 2012.

[21] Chih-Cheng Tseng Haw-Yun Shin Hung-Huan Liu, Wei-Hsiang Lo. A wifi based weighted screening method for indoor positioning system. *wireless Pers Commun*, 79:611–627, 2014.

[22] Mansour Alsaleh-Ahmad Alnafessah Suheer Al-Hadhrami Mai A. Al-Ammar Hend S. Al-Khalifa Abdulrahman Alarifi, AbdulMalik Al-Salman. Ultra wideband indoor positioning technologies: Analysis and recent advances. *Sensors(Basal)*, 16:707, May-2016.

[23] Mohamed R Mahfouz, Cemin Zhang, Brandon C Merkl, Michael J Kuhn, and Aly E Fathy. Investigation of high-accuracy indoor 3-d positioning using uwb technology. *IEEE Transactions on Microwave Theory and Techniques*, 56(6):1316–1330, 2008.

[24] Noelia H Krzysztof Trawinski, Jose M.Alonso. A multiclassifier approach for topology-based wifi indoor localization. *SoftComputing*, 17:1817–1831, 2013.

[25] S. S. Chawathe. Beacon placement for indoor localization using bluetooth. *2008 11th International IEEE Conference on Intelligent Transportation Systems*, pages 980–985, 2008.

[26] You Li Naser El-Sheimy Longning Qi Yuan Zhuang, Jun Yang. Smartphone-based indoor localization with bluetooth low energy beacons. *Sensors*, 16:576, 2016.

[27] Robert de Groote Mortaza S. Bargh. Indoor localization based on response rate of bluetooth inquiries. *MELT*, 2008.

[28] Raffaelo Bruno and Franca Delmastro. Design and analysis of a bluetooth based indoor localization system. *International Federation for Information Processing*, LNCS 2775:711–725, 2003.

[29] Mahamod Ismail Zahid Farid, Rosdiadee Nordin. Recent advances in wireless indoor localization techniques and system. *Journal of Computer Networks and Communications*, page 12, 17th-AUG-2013.

[30] D.Madigan et al. Bayesian indoor positionaing systems. *IEEE INFOCOM*, 2:1217–1227, 2005.

[31] Giovanni Bellusci. Ultra-wideband ranging for low-complexity indoor positioning applications. *Ph.D. Thesis*, 2014.

[32] Uwb localization. `https://wiki.bitcraze.io/doc:lps:index`. Accessed: 2016-10-01.

[33] M. W. Mueller, M. Hamer, and R. D'Andrea. Fusing ultra-wideband range measurements with accelerometers and rate gyroscopes for quadrocopter state estimation. In *Proc. of ICRA*, 2015.

[34] Changzhen Hu Jingfeng Xue Rui Ma, Qiang Guo. An improved wifi indoor positioning algorithm by weighted fusion. *Sensors(basel)*, 15:21834–21843, 2015.

[35] Cheal-Hwan Yoon-Hyunjae Park Jungmin So, Joo-Yub Lee. An improved location estimation method for wifi fingerprint-based indoor localization. *International Journal of Software Applications and its Applications*, Volume-7, 2013.

[36] GuanZhong Ding-Jian Gong-ChenXing Liu Feng Zhao Fan Li, Chunshui Zhao. A reliable and accurate indoor localization method using phone inertial sensors. *UbiComp*, 12, 2012.

[37] Myounggyu Won, Aawesh Shrestha, and Yongsoon Eun. Enabling wifi p2p-based pedestrian safety app. *arXiv preprint arXiv:1805.00442*, 2018.

[38] Myounggyu Won, Shaohu Zhang, and Sang H Son. Witraffic: low-cost and non-intrusive traffic monitoring system using wifi. In *Computer Communication and Networks (ICCCN), 2017 26th International Conference on*, pages 1–9. IEEE, 2017.

[39] Shaohu Zhang, Myounggyu Won, and Sang H Son. Witraffic: Non-intrusive vehicle classification using wifi. In *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD-ROM*, pages 358–359. ACM, 2016.

[40] Myounggyu Won, Taejoon Park, and Sang H Son. Toward mitigating phantom jam using vehicle-to-vehicle communication. *IEEE transactions on intelligent transportation systems*, 18(5):1313–1324, 2017.

[41] Myounggyu Won. A review on v2v communication for traffic jam management. *Vehicle-to-Vehicle and Vehicle-to-Infrastructure Communications: A Technical Approach*, page 1, 2018.

[42] A.Agrawala M.youssef. The horus wlan location determination system. *ACM Mobisys*, pages 205–218, 2005.

[43] Banerjee P-Liu J Liu H, Darabi H. Survey of wireless indoor positioning techniques and systems. *TSMCC*, 37:1067–1080, 2007.

[44] Niemegeers I Gu Y, Lo A. A survey of indoor positioning systems for wireless personnal networks. *IEEE Communication Surveys and Tutorials*, 11:13–32, 2009.

[45] Hyunjae Park-Jungmin So Joo-Yub Lee, Cheal-Hwan Yoon. Analysis of location estimation algorithms for wifi fingerprint-based indoor localization. *SoftTech 2013*, ASTL-19:89–92, 2013.

[46] Pradipta De Quoc Duy Vo. A survey of fingerprint based outdoor localization. *IEEE Communication Surveys and Tutorials*, 18:first quarter, 2016.

[47] Adrian Neild Jean Armstrong Thomas Q Wang, Ahmet Sekercioglu. Position accuracy of time of arrival based ranging using visible light with application in indoor localization systems. *Journal of Lightwave Technology*, 31:3302–3308, 2013.

[48] Leohard Reindl Rui Zhang, Fabian Hoflinger. Inertial sensor based indoor localization and monitoring system for emergency responders. *IEEE-Sensors Journal*, 13, 2013.

[49] Kun Li Qin Lv Robert P.Dick Michael Hannigan Li Shang Yifei Jiang, Xin Pan. Ariel: Automatic wi-fi based room fingerprinting for indoor localization. *Ubiquitous Computing*, 2012.

[50] Venkata N. Padmanabhan Rijurekha Sen Anshul Rai, Krishna kant Chinthalapudi. Zee: Zero-effort crowd sourcing for indoor localization. *In Proceedings of ACM International Conference on Mobile Computing and Networking (MobiCom)-2012*.

[51] Hao Jiang Qingchang Zhu Yeng Chai Soh Zhenghua Chen, Han Zou and Lihua Xie. Fusion of wifi, smartphone sensors and landmarks using the kalman filter for indoor localization. *Sensors*, 15:715 – 732, 2015.

[52] Anthony LaMarca Veljo Otsanon, Alex Varshavsky and Eyal de Lara. Accurate gsm indoor localization. *Ubiquitous Computing*, 3660, 2015.

[53] Ran Liu, Chau Yuen, Tri-Nhut Do, Wei Guo, Xiang Liu, and U-Xuan Tan. Relative positioning by fusing signal strength and range information in a probabilistic framework. pages 1–6. IEEE Globecom Workshops (GC Wkshps)-IEEE, 2016.

[54] Frank Winkler Gunter Fisher, Burkhart Dietrich. Bluetooth indoor localization systems. *Workshop on Positioning, Navigation and Communication*, Volume-4, 2004.

[55] Padmanabhan V Bahl P, Balachandran A. Enhancements to the radar user location and traking system. *Microsoft Research Technical Report*, 2000.

[56] Drafos Niculescu and Badri Nath. Adhoc positioning system (aps) using aoa. *IEEE-INFOCOM*, 2003.

[57] Rong Peng and Mihail L.Sichitiu. Angle of arrival localization for wireless sensor networks. *3rd Annual IEEE Communication Society on Sensor and Ad Hoc Communications and Networks*, pages 374–382, 2006.

[58] Geoffery Messier Carl Wong, Richard Klukas. Using wlan infrastructure for angle-

of-arrival indoor user location. *IEEE- Vehicular Technology Conference*, pages 1–5, 2009.

[59] Guo Zhang Hao He. Localization: Angle-of-arrival. *Computer Networks Lecture:5*, 2017.

[60] Benlarbi-Delai Aziz Julien S.J Quitin Francois Horlin Francois-De Doncker Philippe Vendor Vorst Thomas, Van Eeckhaute Mathieu. Angle-of-arrival based localization using polynomial chaos expansions. *Workshop on Dependable Wireless Communications and Localization for the IOT*, 2017.

[61] Mohammad Heidari and Kaveh Pahlavan. Identification of the absence of direct path in toa-based indoor localization systems. *International Journal of Wireless Information Networks*, 15(3-4):117–127, 2008.

[62] B.Shrivatsava Mani Savvides Andreas, Han Chin Cheih. Dynamic fine-grained localization in ad-hoc networks of sensors. *Annual International Conference on Mobile Computing and Networking*, 2002.

[63] Christine L.Julien XiaoWei Luo, William J.O'Brien. Comparative evaluation of received signal-strength index (rssi) based indoor localization for construction jobsites. *Mobile and Pervasive Computing Group- Advanced Engineering Information*, 2010.

[64] Crazyflie 2.0. `https://wiki.bitcraze.io/doc:lps:lpp#two_way_ranging`. Accessed: 2016-03-01.

[65] Neil F.Jhonson Pedro D.Manrique, D.Dylan Jhonson. Using competition to control congestion in autonomous drone systems. *Electronics*, 6:31, 2017.

[66] D. Giordan, A. Manconi, F. Remondino, and F. Nex. Use of unmanned aerial vehicles in monitoring application and management of natural hazards. *Geomatics, Natural Hazards and Risk*, 8(1):1–4, 2017.

[67] Crazyflie 2.0. `https://www.bitcraze.io/crazyflie-2/`. Accessed: 2016-03-01.

[68] Esp8266 serial wifi module. `https://www.itead.cc/wiki/ESP8266_Serial_WIFI_Module#AT_Commands`. Accessed: 2016-05-01.

[69] Esp8266 integration with arduino uno module. `http://www.teomaragakis.com/hardware/electronics/how-to-connect-an-esp8266-to-an-arduino-uno/`. Accessed: 2016-05-01.

[70] Nora Ayanian Wolfgang Honig. Flying multiple uav's using ros. pages 83–118, 2017.

[71] Karthik Dantu Matt Welsh Bryan Kate, Jason Waterman. Simbeeotic: A simulator and testbed for micro-aerial vehicle swarm experiments. pages 49–60. Information Processing in Sensor Networks (IPSN), 2012.

[72] Find localization system. `https://doc.internalpositioning.com/`. Accessed: 2016-10-01.

[73] Martin Kaltenbrunner, Getachew Adam, Eric Daniel Głowacki, Michael Drack, Reinhard Schwödiauer, Lucia Leonat, Dogukan Hazar Apaydin, Heiko Groiss,

Markus Clark Scharber, Matthew Schuette White, et al. Flexible high power-per-weight perovskite solar cells with chromium oxide-metal contacts for improved stability in air. *Nature materials-Nature Publishing Group*, 14(10):1032–1039, 2015.